Computers Math. Applic. Vol. 18, No. 9, pp. 823–834, 1989
Printed in Great Britain. All rights reserved

0097-4943/89 $3.00 + 0.00
Copyright © 1989 Maxwell Pergamon Macmillan plc

PARTICLE SIMULATION OF BIOLOGICAL SORTING ON A SUPERCOMPUTER†

D. GREENSPAN
Mathematics Department, University of Texas at Arlington, Arlington, TX 76019, U.S.A.

Abstract—Cell sorting, or self reorganization, is modelled by means of particles which obey classical molecular dynamical equations. A system of N second order, nonlinear, ordinary differential equations results when the number of particles is N. Applications and examples are described and discussed, with $N > 1000$, for both double layer and triple layer self reorganization.

1. INTRODUCTION

Biological cell sorting, or, self reorganization, has received wide attention in recent years (see, e.g., Refs [1–5] and the numerous references contained therein). Motivation is derived from biological experiments in which separated tissues self reorganize.

A popular approach to the mathematical modelling of cell sorting is that of Steinberg [5], which is founded on cellular motility and differential adhesion hypotheses, which, in turn, are dominated by the principle of minimization of free energy. Computer implementation of this approach has been explored using diverse dynamical possibilities, including special motility rules, extended zone effects and interface tension. In all related computer simulations, the fundamental problem considered was as follows. In the plane, consider a collection of cells which consists of two different types, say, A and B. The initial positions of the cells are selected at random within a square. Confining all motions to be within the given square and using only *local* cell interaction rules, one must induce the A cells to organize into a central, relatively circular core, while the B cells form a layer around this core.

In this paper we will modify Steinberg's theory in a natural and suitable fashion [2], so that self reorganization follows readily in the desired fashion. The free energy minimization principle will be replaced by the mechanics of classical molecular interaction. CRAY X-MP/24 computer examples for both double and triple layer self reorganization will be described and discussed. The model is relatively simplistic in that it does not incorporate chemical or biological principles directly, but uses laws of physics.

2. CLASSICAL MOLECULAR MECHANICS

For purposes of intuition, it will be important to review, first, how molecules interact. Within a larger body, molecules interact only locally, that is, only with their nearest neighbors. This interaction is of the following general nature [6]. If two molecules are pushed together they repel each other, if pulled apart they attract each other, and mutual repulsion is of a greater order of magnitude than is mutual attraction. Mathematically, this behavior is often formulated as follows [7]. The magnitude F of the force $\mathbf{F}$ between two molecules which are locally r units apart is of the form

$$F = -\frac{G}{r^p} + \frac{H}{r^q}. \tag{1}$$

where, typically,

$$G > 0, \quad H > 0, \quad q > p \geqslant 7. \tag{2}$$

†Computations performed at the University of Texas Center for High Performance Computing.

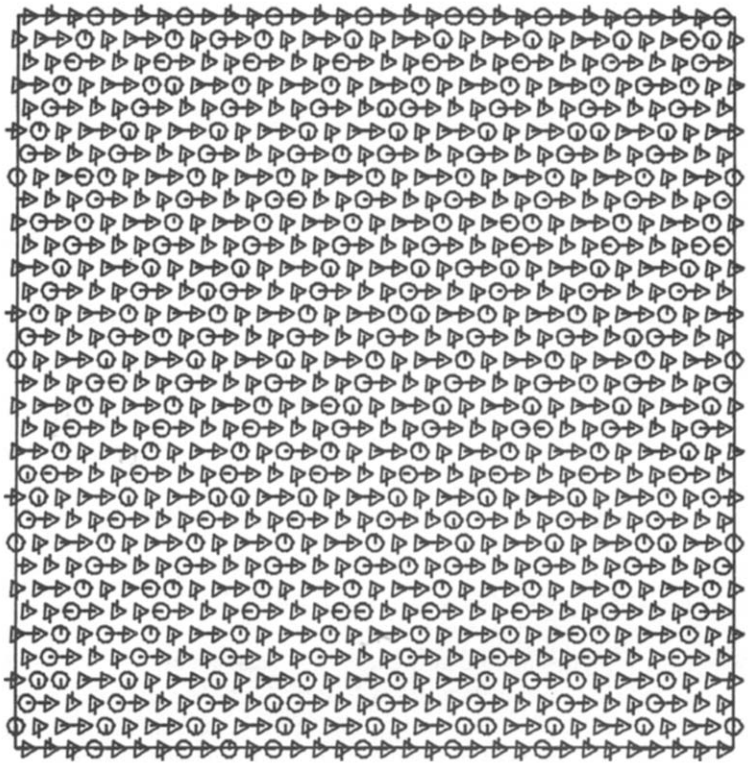

Fig. 1. Initial data.

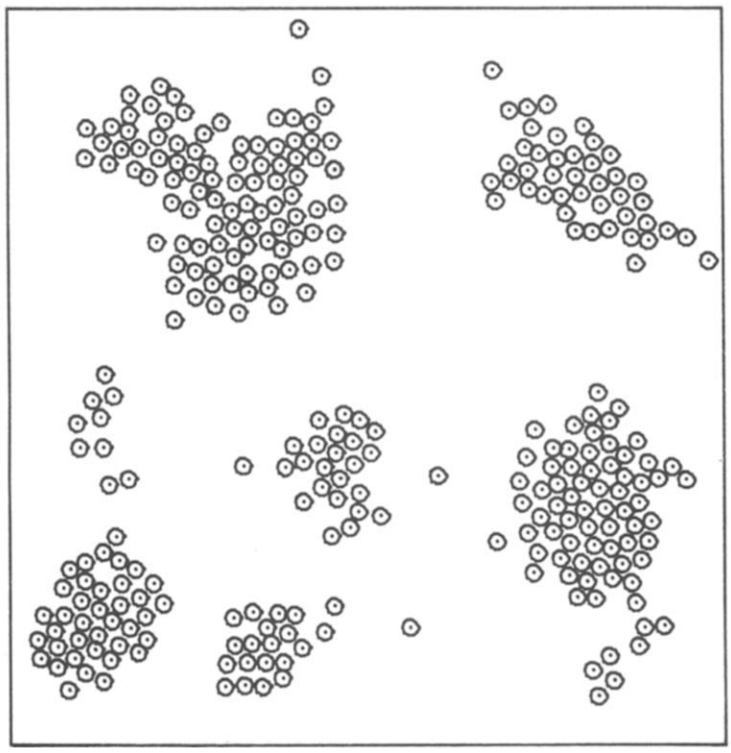

Fig. 2. $T = 0.5$.

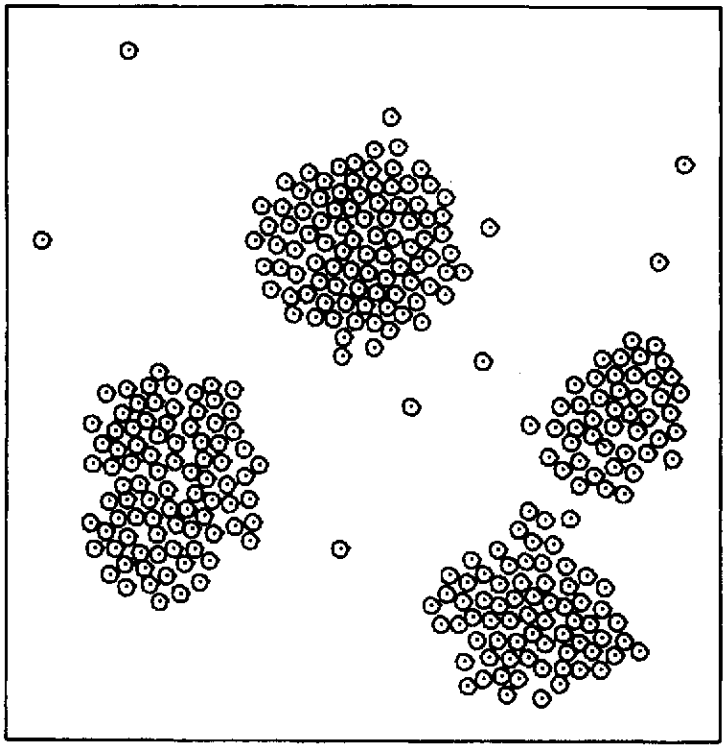

Fig. 3. $T = 1.5$.

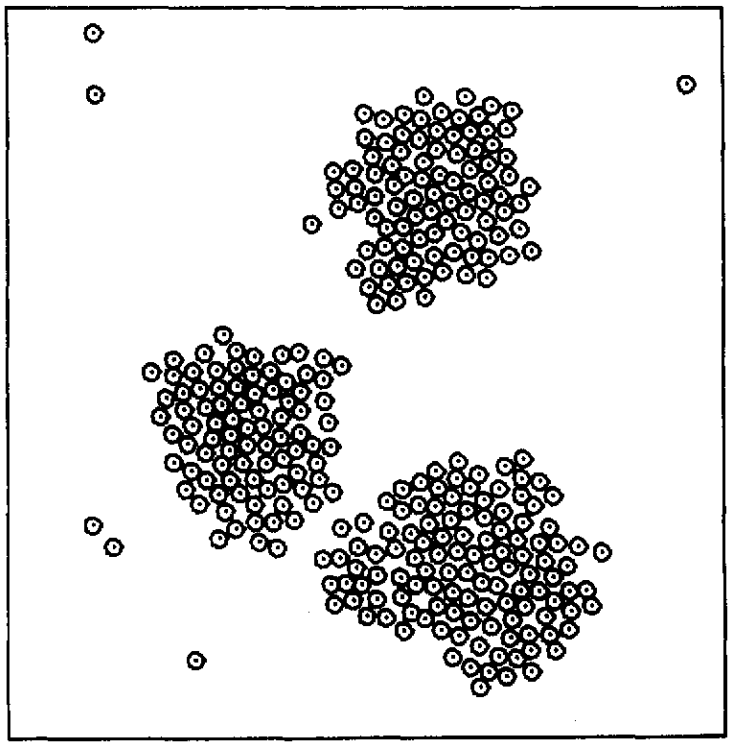

Fig. 4. $T = 2.5$.

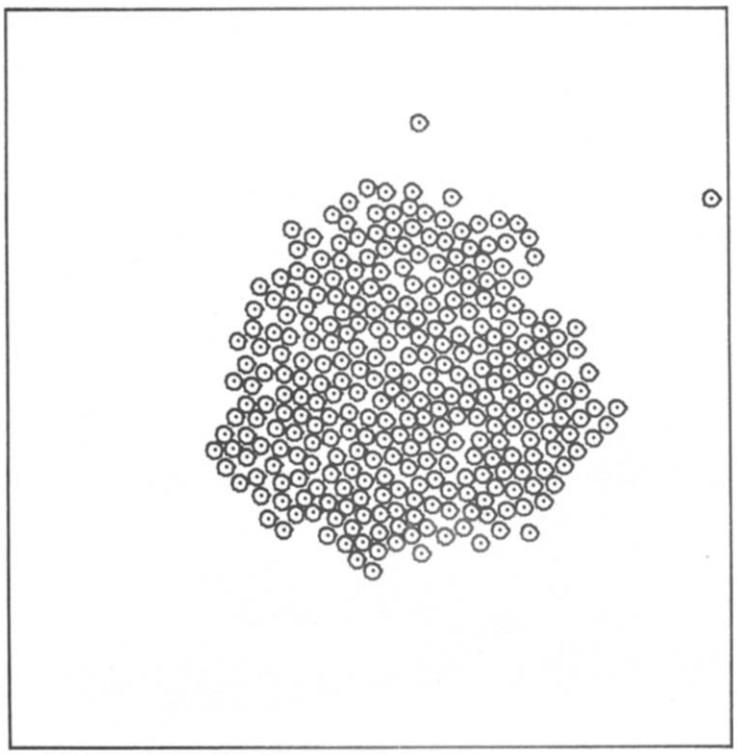

Fig. 5. $T = 4$.

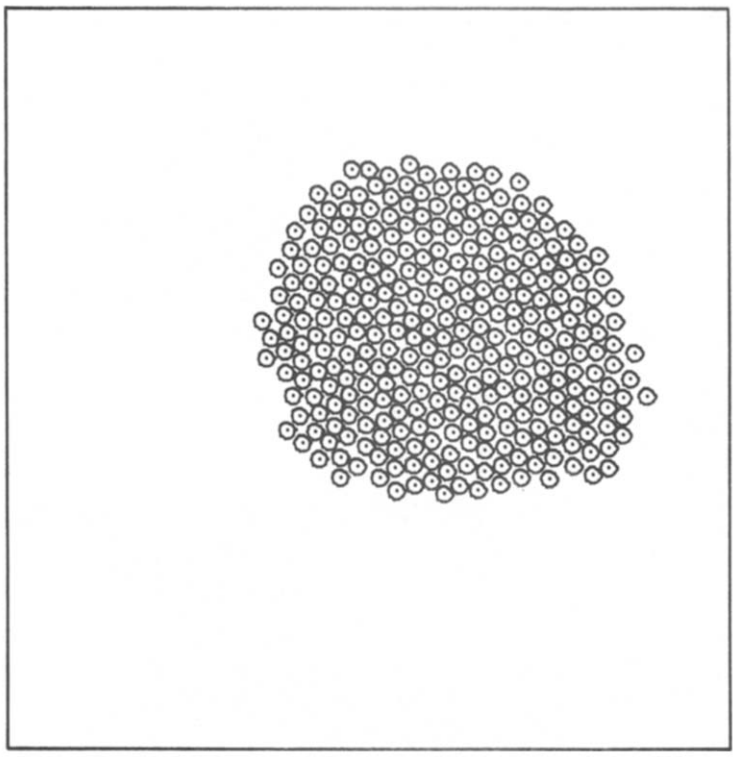

Fig. 6. $T = 9$.

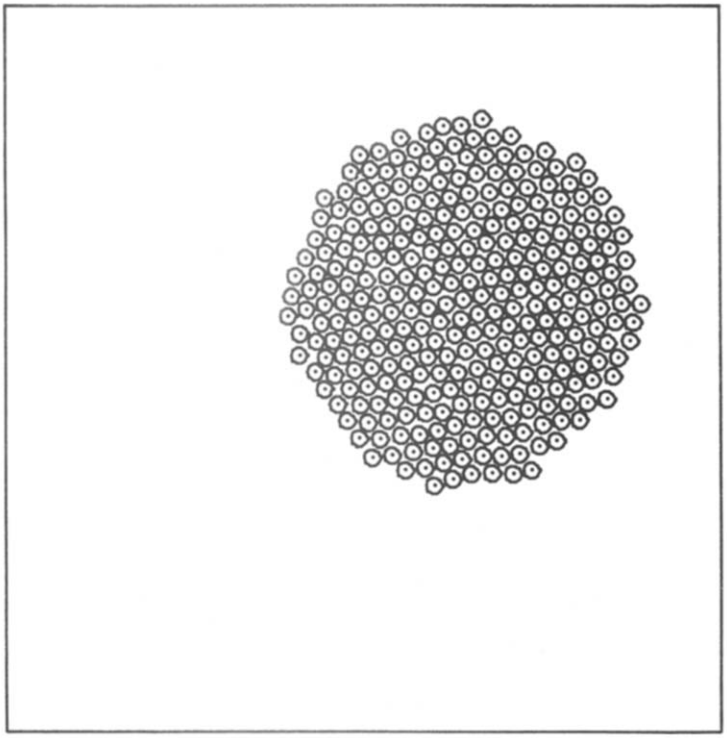

Fig. 7. $T = 16.5$.

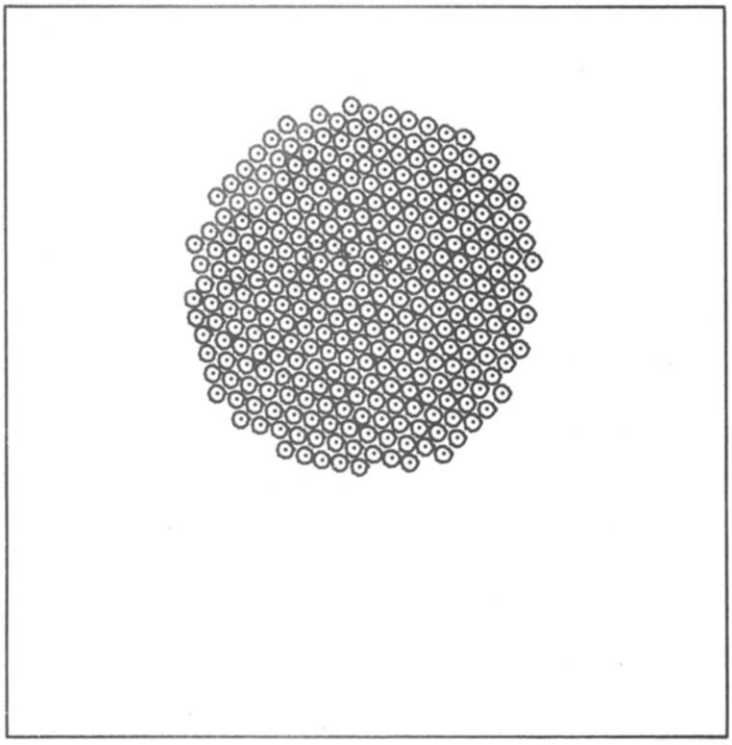

Fig. 8. $T = 24$.

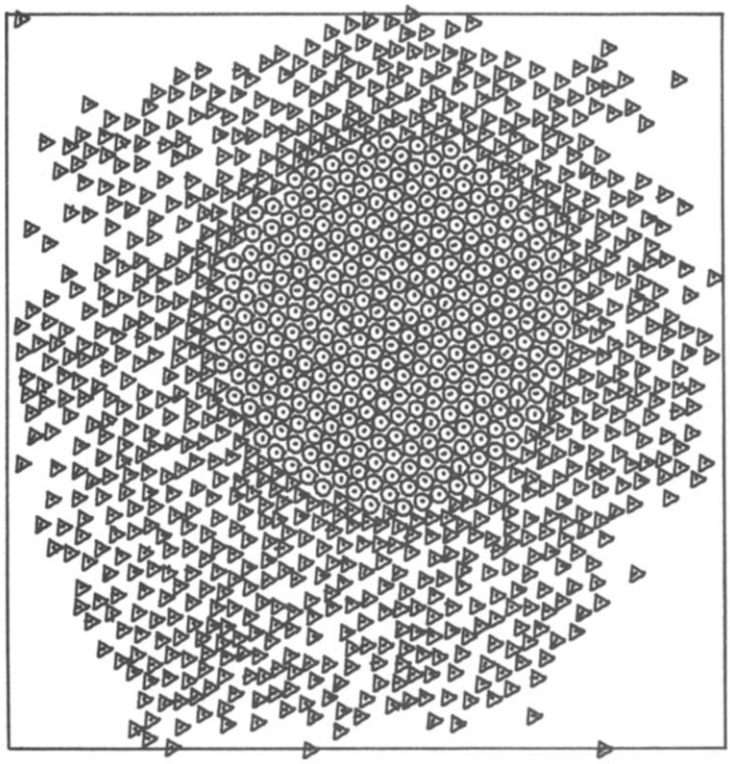

Fig. 9. $T = 31.5$.

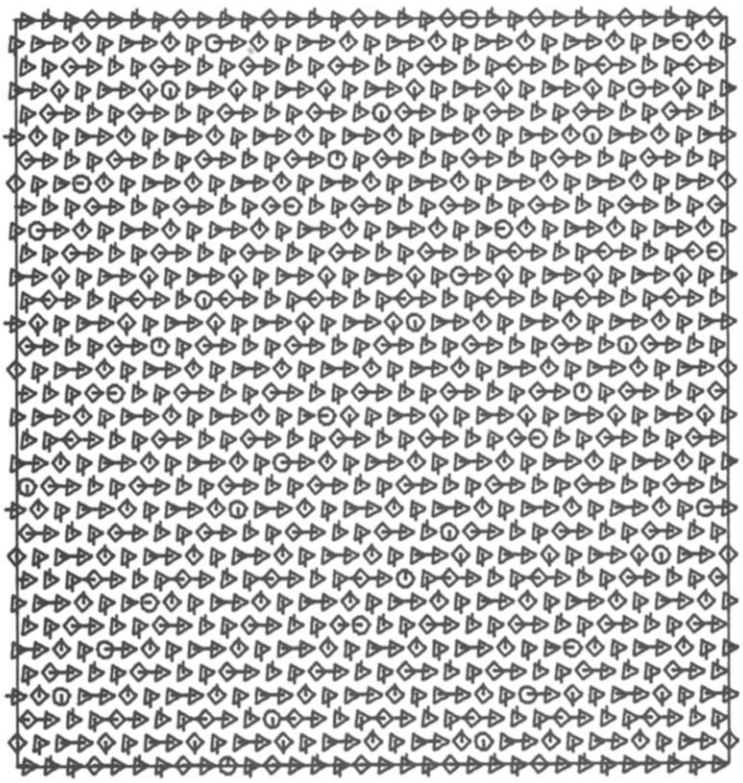

Fig. 10. Initial data.

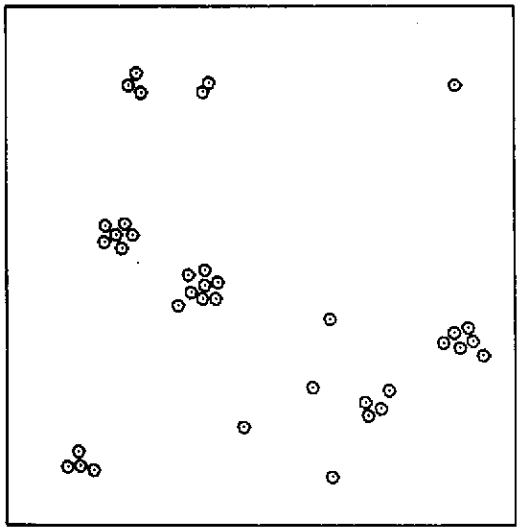

Fig. 11. $T = 1.5$.

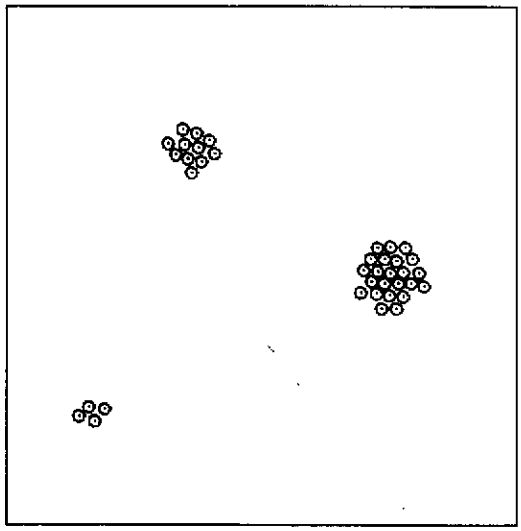

Fig. 12. $T = 9$.

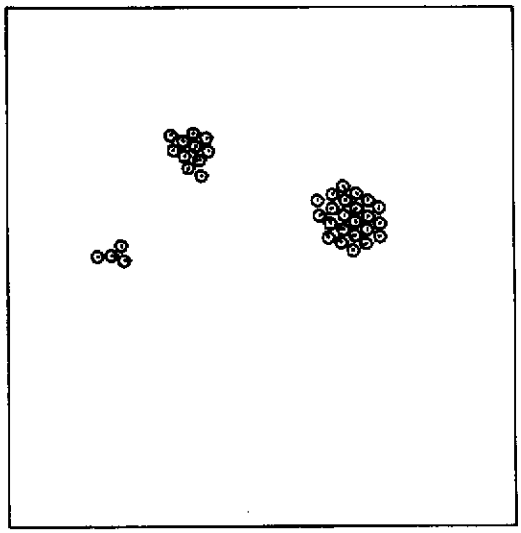

Fig. 13. $T = 16.5$.

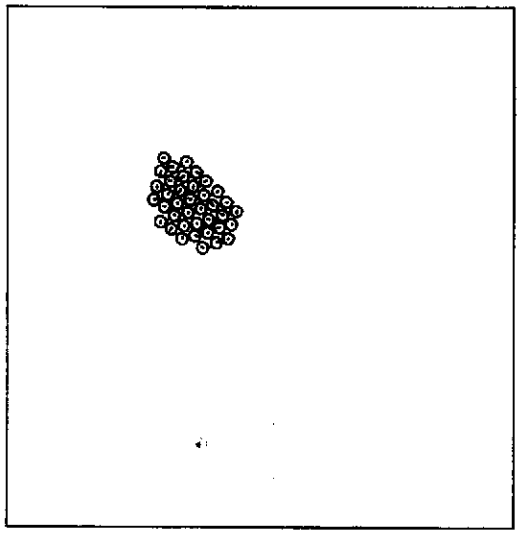

Fig. 14. $T = 24$.

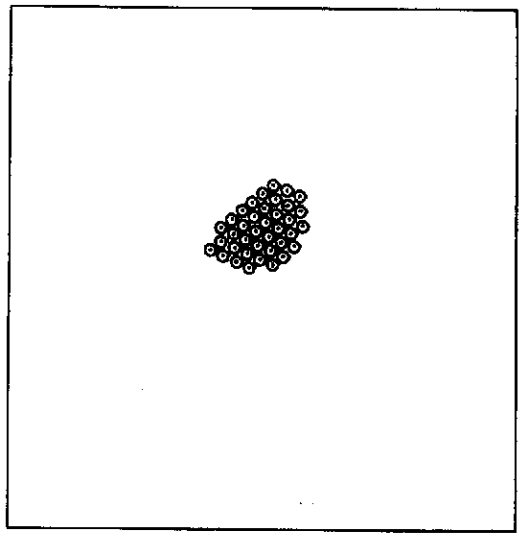

Fig. 15. $T = 31.5$.

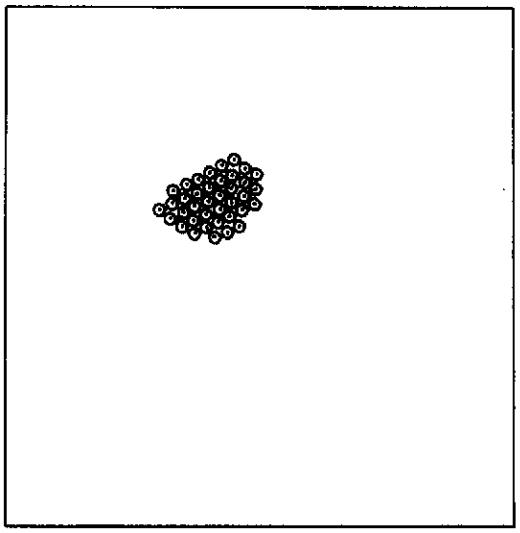

Fig. 16. $T = 39$.

The major problem in any simulation of a physical body is that there are too many component molecules to incorporate into the model. The classical mathematical approach is to replace the large, but finite, number of molecules by an infinite set of points. In so doing, the rich physics of molecular interaction is lost because every point always has an infinite number of neighbors which are arbitrarily close. A viable computer alternative is to replace the large number of *molecules* by a much smaller number of *particles* and then readjust the parameters in equation (1) to compensate [8]. It is this latter approach which we will follow.

3. THE COMPUTER ALGORITHM

The general idea outlined in Section 2 will be implemented in the following, constructive fashion. Consider N particles P_i, $i = 1, 2, \ldots, N$. For $\Delta t > 0$, let $t_k = k\,\Delta t$, $k = 0, 1, 2, \ldots$. For each of $i = 1, 2, \ldots, N$, let m_i denote the adhesive measure of P_i, and let P_i at t_k be located at $\mathbf{r}_{i,k} = (x_{i,k}, y_{i,k}, z_{i,k})$, have velocity $\mathbf{v}_{i,k} = (v_{i,k,x}, v_{i,k,y}, v_{i,k,z})$, and have acceleration $\mathbf{a}_{i,k} = (a_{i,k,x}, a_{i,k,y}, a_{i,k,z})$. Let position, velocity, and acceleration be related by the recursion formulas

$$\mathbf{v}_{i,1/2} = \mathbf{v}_{i,0} + \tfrac{1}{2}(\Delta t)\mathbf{a}_{i,0} \quad \text{(starter formula)} \tag{3}$$

$$\mathbf{v}_{i,k+1/2} = \mathbf{v}_{i,k-1/2} + (\Delta t)\mathbf{a}_{i,k}, \quad k = 1, 2, 3, \ldots, \tag{4}$$

$$\mathbf{r}_{i,k+1} = \mathbf{r}_{i,k} + (\Delta t)\mathbf{v}_{i,k+1/2}, \quad k = 0, 1, 2, \ldots. \tag{5}$$

At t_k, let the force acting on P_i be $\mathbf{F}_{i,k}$. We relate force and acceleration by the dynamical equation

$$\mathbf{F}_{i,k} = m_i \mathbf{a}_{i,k}. \tag{6}$$

As soon as the precise structure of $\mathbf{F}_{i,k}$ is given, the motion of each P_i will be determined explicitly and recursively by equations (3)–(6) from given initial data. The force $\mathbf{F}_{i,k}$ is described now as follows. Let $\mathbf{r}_{ij,k}$ be the vector from P_i and P_j at time t_k, so that $r_{ij,k} = |\mathbf{r}_{i,k} - \mathbf{r}_{j,k}|$ is the distance between the two particles. Then the force $\mathbf{F}_{ij,k}$ on P_i exerted by P_j at time t_k is assumed to be

$$\mathbf{F}_{ij,k} = \left(\left(-\frac{G_{ij}}{(r_{ij,k})^p} + \frac{H_{ij}}{(r_{ij,k})^q}\right)\frac{\mathbf{r}_{ji,k}}{r_{ij,k}}\right), \tag{7}$$

which is consistent with equation (1). The total force $\mathbf{F}_{i,k}$ on P_i due to all other particles different from P_i is defined by

$$\mathbf{F}_{i,k} = \sum_{\substack{j=1 \\ j \neq i}}^{N} \left\{\left\{-\frac{G_{ij}}{(r_{ij,k})^p} + \frac{H_{ij}}{(r_{ij,k})^q}\right\}\frac{\mathbf{r}_{ji,k}}{r_{ij,k}}\right\}. \tag{8}$$

Note finally that the introduction of an additional parameter D is esential to assure that particle interactions are local. We will require that whenever $r_{ij,k} > D$, then equation (7) must be replaced with

$$\mathbf{F}_{ij,k} = \mathbf{0} \quad (r_{ij,k} > D). \tag{9}$$

Note that the vector form of equations (3)–(9) makes them applicable in any number of dimensions.

For the convenience of the reader, a typical CRAY X-MP/24 FORTRAN program of the algorithm described above is available in the Appendix of Greenspan [9].

4. COMPUTER EXAMPLES

Since, in general, particles do not adhere when in a gaseous state and are rigid when in a solid state, sorting can only occur in a liquid, or near-liquid, state. Relative to this observation, the

following preliminary remarks will be important. For the time step $\Delta t = 0.0001$, in equation (7) set

$$p = 3, \quad q = 5 \tag{10}$$

and

$$G_{ij} = H_{ij} = 5m_i m_j. \tag{11}$$

Let the local distance of interaction parameter be given by

$$D = 2.2. \tag{12}$$

Then, if P_i is to be a liquid particle, the range of the speed v_i of P_i can be deduced for various adhesive measures m_i. In particular [8]:

$$m_i = 2000, \text{ implies } 100 \leqslant v_i \leqslant 170, \tag{13}$$

$$m_i = 4000, \text{ implies } 90 \leqslant v_i \leqslant 160, \tag{14}$$

$$m_i = 10{,}000, \text{ implies } 50 \leqslant v_i \leqslant 80. \tag{15}$$

The parameter choices throughout this section will be guided by (10)–(15).

Let us now examine the following detailed example. Consider the square region in the XY plane whose vertices are (16, 16), (−16, 16), (−16, −16), (16, −16). In this region, construct a triangular mosaic of 1072 grid points in the following fashion, which, incidentally, is sufficiently general to allow the construction of both larger and smaller sets of such grid points. Set

$$x_1 = -25.0, \quad y_1 = 25.0, \quad x_{52} = -24.5, \quad y_{52} = 24.0,$$

$$x_{i+1} = 1 + x_i, \quad y_{i+1} = 25.0; \quad i = 1, 2, \ldots, 50,$$

$$x_{i+1} = 1 + x_i, \quad y_{i+1} = 24.0; \quad i = 52, 53, \ldots, 100,$$

$$x_i = x_{i-101}, \quad y_i = -1 + y_{i-101}; \quad i = 102, 103, \ldots, 2576.$$

The resulting 2576 points P_i, with respective coordinates (x_i, y_i), are the vertices of a triangular mosaic which fills the square whose vertices are (25, 25), (−25, 25), (−25, −25), (25, −25). To determine the 1072 such points which lie within and on the square whose vertices are (16, 16), (−16, 16), (−16, −16), (16, −16), we merely exclude those of the 2576 points which satisfy any one of $x_i > 16$, $x_i < -16$, $y_i > 16$, $y_i < -16$.

Next, we fix the set A to have 304 particles, each with $m_i = 10{,}000$, and the set B to have 768 particles, each with $m_i = 2000$. Each particle is set at a distinct grid point as shown in Fig. 1, where the particles of set A are represented by circles, while those of set B are represented by triangles. The particles of set A have been distributed widely throughout the square.

Next, a velocity is assigned to each particle, by a random process, in one of the four directions N, S, E, W. For each particle in B, the speed is 150. For each particle in A, the speed is either 50 or 80, determined at random. The velocity of each particle is shown in Fig. 1 as a vector emanating from each particle's center. All initial data are now assigned. For a complete listing, see the Appendix of Greenspan [9].

Now, fix $\Delta t = 0.0001$ and let the system parameters be given by equations (10)–(12). In order to keep the particles within the square while they are in motion, the following reflection rules are applied:

(a) if $x_i > 16$, reset $x_i \to 32 - x_i$, $v_{x,i} \to -0.99 v_{x,i}$, $v_{y,i} \to 0.99 v_{y,i}$,

(b) if $x_i < -16$, reset $x_i \to -32 - x_i$, $v_{x,i} \to -0.99 v_{x,i}$, $v_{y,i} \to 0.99 v_{y,i}$,

(c) if $y_i > 16$, reset $y_i \to 32 - y_i$, $v_{x,i} \to 0.99 v_{x,i}$, $v_{y,i} \to -0.99 v_{y,i}$,

(d) if $y_i < -16$, reset $y_i \to -32 - y_i$, $v_{x,i} \to 0.99\, v_{x,i}$, $v_{y,i} \to -0.99 v_{y,i}$.

The small amount of velocity damping in rules (a)–(d) insures the stability of equations (3)–(9) when using $\Delta t = 0.0001$, which will be discussed in greater detail later.

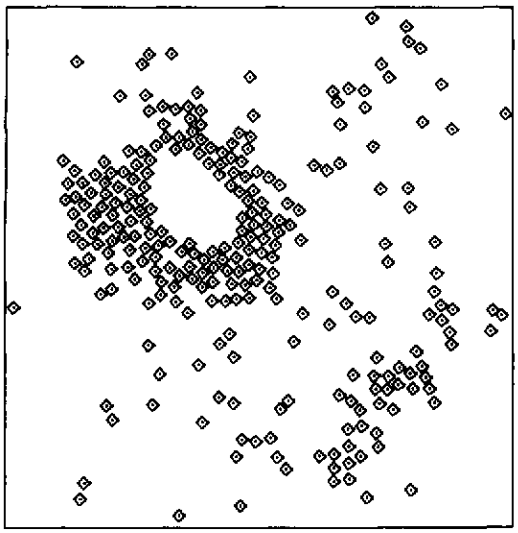

Fig. 17. $T = 24$.

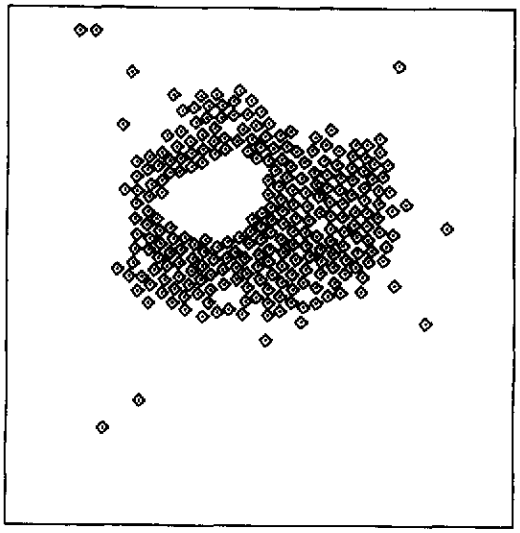

Fig. 18. $T = 31.5$.

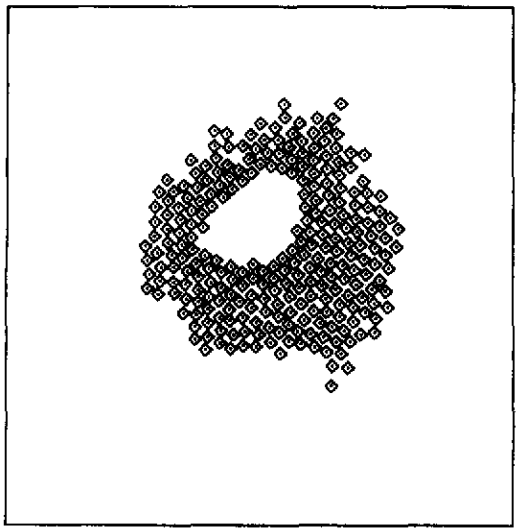

Fig. 19. $T = 39$.

Figures 2–5 show the self reorganization process of the A particles at the respective times $T = 0.5$, 1.5, 2.4, 4.0. They reveal that these particles first form into small subsets, which then form into larger subsets, and finally form a central core. Figures 6–8, at the respective times $T = 9.0$, 16.5, 24.0, show the self reorganization of the set A into a relatively circular core. At this point, it was observed that the formation of an outer layer by the B particles was a relatively slow process. Hence, as an economy move, we reduce the damping factor in rules (a)–(d) to 0.9 after $T = 24.0$. Figure 9 then shows, at $T = 31.5$, the circular central core of A particles and the layer of B particles around the core.

With regard to execution times on the CRAY, 1000 time steps required 48 s of cpu time. Thus, the total cpu time until $T = t_{315,000} = 31.5$ was 4.2 h.

As a second example, let us consider a set which consists of three different types of particles A, B and C, and show how to induce a self reorganization in which the A particles form a central core, the B particles form a layer around the A particles, and the C particles form a layer around the B particles. The biological analog would be the self reorganization into normal tissue of separated endoderm, mesoderm and ectoderm cells. All considerations are the same as in the first example with the following exceptions. The sets A, B, C have 38, 266 and 768 particles, respectively, with adhesion measures 10000, 4000 and 2000, respectively. The particles are positioned within the

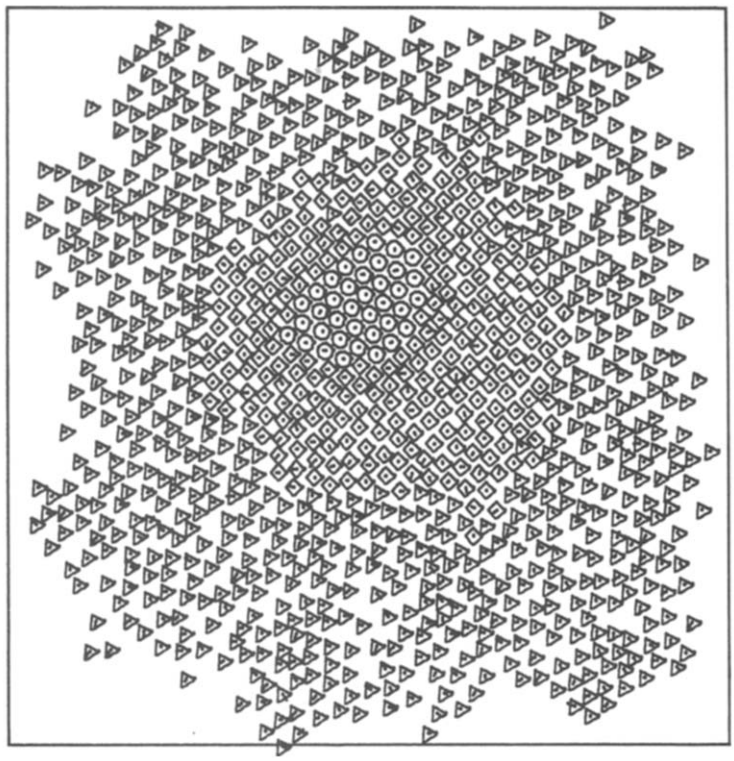

Fig. 20. $T = 39$.

32×32 square so that the A and B sets have been widely separated. The A particles are each assigned speeds of 60 while the B and C particles are assigned speeds of 150. The initial data are displayed in Fig. 10, with the A particles represented by circles, the B particles by quadrilaterals, and the C particles by triangles. A complete initial data listing is given in the Appendix of Greenspan [9].

The resulting self reorganization is shown in Figs 11–20. Figures 11–16 show the self reorganization of the A cells at the respective times $T = 1.5, 9.0, 16.5, 24.0, 31.5, 39.0$. Figures 17–19 show the self reorganization of the B particles around the core at the respective times $T = 24.0$, 31.5, 39.0. Figure 20 shows the triple self reorganization of the A, B, C sets at time $T = 39.0$. The exceptionally slow self reorganization of sets B and C, after the A particles formed into the core, was, again, accelerated by setting damping factor 0.99 to 0.9 in rules (a)–(d) after $T = 24.0$.

5. REMARKS

Though small parameter variations in the two examples in Section 3 yielded entirely analogous results, large variations often did not. From the large number of additional examples run, we now discuss some of the types of problems which were thereby encountered.

In the first example of Section 3, changing D to 3.0 and eliminating all damping resulted in successful execution until, approximately, $t_{40,000}$. The result at this time was entirely analogous to that shown in Fig. 5. However, allowing D to be 3 increased the force after $t_{40,000}$ on each particle in the now highly concentrated A set to the point that overflow resulted after $t_{40,000}$. There were three possible remedies to correct this numerical instability. First, one could decrease D. Second, one could introduce a small amount of damping in order to decrease particle velocities, and hence decrease the system's total kinetic energy, or temperature. Third, one could decrease the time step. Since decreasing the time step was unfeasible economically, both the first and second remedies were implemented, which is, in fact, how the parameters of the first example were actually determined.

Next note that if one allows the adhesion constants m_i to be "close", like 10,000 and 9500, then self reorganization is an exceptionally slow process.

Note also that if, *from the start*, one chooses the damping factor in rules (a)–(d) to be too small, for example, 0.9, then trapping can result, that is, a particle from set B can always be found interior to set A in both the examples of Section 3. The reason is that there follows an excessive loss of system kinetic energy, and this yields premature solidification. The biological implication is that sorting can occur only above a certain cell temperature T_1, which is characteristic of the cells under consideration.

Finally, note that for $D < 1.4$ and *without* damping, we were not able to achieve sorting. This may have been due to the time constraints, or due to a system kinetic energy level which was too high and yielded particle behavior like that of a gas, rather than that like a liquid. If the second possibility is correct, then there would also exist a temperature T_2, which is characteristic of the cells under consideration, *above* which sorting cannot result, and hence, sorting could only occur within the temperature range $T_1 < T < T_2$.

REFERENCES

1. P. Antonelli, T. D. Rogers and M. Willard, Cell aggregation kinetics. *J. theor. Biol.* **41**, 1–21 (1973).
2. D. Greenspan, A classical molecular approach to computer simulation of biological sorting. *J. Math. Biol.* **12**, 227–235 (1981).
3. R. J. Matela and R. J. Fletterick, Computer simulation of cellular self-sorting. *J. theor. Biol.* **84**, 673–690 (1980).
4. G. Rogers and N. S. Goel, Computer simulation of cellular movements: cell-sorting, cellular migration through a mass of cells and contact inhibition. *J. theor. Biol.* **71**, 141–166 (1978).
5. M. S. Steinberg, Reconstructing of tissues by dissociated cell. *Science* **141**, 401–408 (1963).
6. R. P. Feynman, R. B. Leighton and M. Sands, *The Feynman Lectures on Physics*. Addison-Wesley, Reading, Mass. (1963).
7. J. O. Hirschfelder, C. F. Curtiss and R. B. Bird, *Molecular Theory of Gases and Liquids*. Wiley, New York (1954).
8. D. Greenspan, *Arithmetic Applied Mathematics*. Pergamon Press, Oxford (1980).
9. D. Greenspan, Particle simulation of biological sorting on a supercomputer. Technical Report 254. Department of Mathematics, University of Texas at Arlington, Tex. (1988).